Cambridge Centre for Computational Chemical Engineering

University of Cambridge

Department of Chemical Engineering

User Manual for SWEEP 2: Population Balance Software.

Matthew Celnik

submitted: May 18, 2006

c4e

Edited by
Cambridge Centre for Computational Chemical Engineering
Department of Chemical Engineering
University of Cambridge
Cambridge CB2 3RA
United Kingdom.

Fax: + 44 (0)1223 334796
E-Mail: c4e@cheng.cam.ac.uk
World Wide Web: http://www.cheng.cam.ac.uk/c4e/

Contents

1 Introduction **3**

2 File List **4**

3 Compiling the Code **4**

4 Running the Program **5**

5 Physical Model **5**

5.1 Inception reactions 6

5.2 Surface Reactions 6

5.3 Condensation reactions 7

5.4 Coagulation 7

5.5 Ensemble Rates 8

6 Particle Models **8**

7 Settings Input File **8**

7.1 CHEMFILE keyword 8

7.2 MECHFILE keyword 9

7.3 RUNS keyword 9

7.4 STARTT keyword 9

7.5 TIME keyword 9

7.6 PCOUNT keyword 9

7.7 MAXM0 keyword 10

7.8 ACTSURFMODEL keyword 10

7.9 OUTPUTFILE keyword 10

7.10 PMODEL keyword 10

7.11 LOWSTAT keyword 11

7.12 HIGHSTAT keyword 11

8 Mechanism Input File (XML) 11
8.1 mechanism element . . . 11
8.2 model element . . . 11
8.2.1 Accepted model elements . . . 12
8.3 particle element . . . 12
8.4 component element . . . 12
8.5 density element . . . 12
8.6 molwt element . . . 13
8.7 track element . . . 13
8.8 inception element . . . 13
8.9 reaction element . . . 14
8.9.1 surface type reactions . . . 15
8.9.2 condensation type reactions . . . 15
8.10 formula element . . . 16
8.11 reactant element . . . 16
8.12 product element . . . 16
8.13 particlein element . . . 16
8.14 particleout element . . . 17
8.15 A, n and E elements . . . 17
8.16 particleterm element . . . 17
8.17 coagulation element . . . 17

9 Chemistry Profile File 18

10 Output Files 19

11 Example Settings File 20

12 Example Chemistry Profile 20

13 Example Mechanism File 20

1 Introduction

SWEEP is a population balance solver for particulate systems which exist dispersed in a gas-phase. It is based on a Monte-Carlo method and includes various algorithmic enhancements. This manual briefly outlines the functionality of the software and provides all information required for a user to run the program.

The internals of the population balance simulation are a subject of current research and are not described here, reference should be made to the original papers, which are listed at http://como.cheng.cam.ac.uk.

Please cite this software as SWEEP2: Cambridge Soot Simulator, Cambridge, 2006, http://como.cheng.cam.ac.uk.

Sweep is the name of a dog from a childrens TV show in which the main character, a bear, is called Sooty.

2 File List

sweep.f90	Main source file of SWEEP library, used by calling code.
mt19937.f	An implementation of the Mersenne twister random number generator.
profile_driver_input.f90	Input routines for driver program.
profile_driver_output.f90	Output routines for driver program.
profile_driver_program.f90	The driver program for profiled chemistry data.
swpchem_point.f90	Chemistry module for variable chemistry expressed as a single array of gas-phase variables.
swpchem_point_dppt.f90	Same as swpchem_point.f90 except the chemistry is stored in double precision and accessed via a pointer.
swpchem_profile.f90	Chemistry routines for fixed chemistry expressed as a profile of gas-phase variables at different time points.
swpchem_shared.f90	Routines and data shared by all flavours of swpchem_*.f90.
swpcoag.f90	Routines for calculating the rates of particle coagulation.
swpcoag_model.f90	Definition of the coagulation model, including kernels for different flow regimes.
swpensemble.f90	Definition of the particle ensemble type and routines which act upon it.
swperr.f90	List of error codes produced by SWEEP and their descriptions.
swpmech.f90	Routines which set and read the mechanism type.
swpmech_reader.f90	Routines which read a mechanism from an XML file.
swpmech_types.f90	Definitions of the mechanism type and its dependent types.
swparams.f90	List of control parameters and some constants used by SWEEP.
swppart.f90	Definition of the particle type and routines which act upon it.
swpprocess.f90	Routines which calculate process rates and perform both DSA and LPDA processes on the particle ensemble.
swprng.f90	Wrapper for the random number generator currently in use (mt19937.f).
swpsoln.f90	Definition of the solution type and routines which act upon it.
swpstats.f90	Routines for extracting useful statistics from a particle ensemble and compiling particle lists for export.
swpstep.f90	The stochastic stepping algorithm routines.

3 Compiling the Code

The code was compiled on Win32 systems using the Compaq Visual Fortran (CVF) 6.6 IDE. Workspace and project files are supplied for this purpose. For Linux systems a

Makefile is provided which compiles the program using the Intel fortran compiler (ifort).

4 Running the Program

SWEEP is provided with a driver program which reads the gas-phase chemistry profile from a text file (tab separated or CSV) and solves the population balance without coupling to the gas-phase. As SWEEP was developed as a library of routines there is no obligation to use this program, though it should be a useful starting point for writing customised driver programs.

Three input files are required to run SWEEP with this driver program:

1. A settings input file (sweep.inp).
2. The chemistry profile file.
3. The particle mechanism file in XML format.

The names of the chemistry profile file and the mechanism file are given in the settings file, see section 7 for more details. The structure of these input files is detailed in subsequent sections. To run the program collect all the input files together with the executable in a single directory and execute.

5 Physical Model

SWEEP solves the Smoluchowski population balance equation using a Monte-Carlo particle technique with the Linear Process Deferment Algorithm (LPDA) enhancement (Patterson *et al.*, 2006a). The particle population is modelled in a gas-phase environment and soot processes can also include gas-phase species interactions. The processes which can be modelled by SWEEP are:

1. Particle inception (zero particle events).
2. Surface reactions (single particle events).
3. Surface condensations (single particle events).
4. Coagulation (two particle event).

5.1 Inception reactions

Inception reactions in SWEEP are modelled as two-body species collisions. The collision is modelled using the transition kernel of Patterson *et al.* (2006b). The rate (per unit volume) takes the form:

$$R_{inception} = \frac{1}{2} k_{tr} N_A^2 C_A C_B \tag{1}$$

Where k_{tr} is the transition regime coagulation kernel constant, N_A is Avodgadro's number, C_A is the gas-phase concentration of the first incepting species and C_B is the gas-phase concentration of the second incepting species. The constants for the free-molecular kernel (k_{fm}), the slip-flow kernel (k_{sf}) and the transition kernel (k_{tr}) are defined as:

$$k_{fm} = 2.2\sqrt{\frac{\pi k_B T}{2}} \left(\frac{1}{m_A} + \frac{1}{m_B} \right)^{\frac{1}{2}} (d_A + d_B)^2 \tag{2}$$

$$k_{sf} = \frac{2k_b T}{3\mu} \left(\frac{1 + 1.257 Kn_A}{d_A} + \frac{1 + 1.257 Kn_B}{d_B} \right) (d_A + d_B) \tag{3}$$

$$k_{tr} = \frac{k_{fm} k_{sf}}{k_{fm} + k_{sf}} \tag{4}$$

k_B is the Boltzmann constant, T is the temperature and μ is the viscosity of the gas-phase. m_i and d_i are the mass and collision diameter of species i respectively. The Knudsen number of a sphere of diameter d_i is given by:

$$Kn_i = 4.74 \times 10^{-8} \frac{T}{P\, d_i} \tag{5}$$

Where P is the pressure.

5.2 Surface Reactions

Surface reactions rates (per particle) in SWEEP have the general form:

$$R_{surface} = A T^n \exp\left(-\frac{E}{RT} \right) \prod_{j=1}^{J} \theta_j^{p_j} \prod_{i=1}^{I} C_i^{\nu_i} \tag{6}$$

Where A, n and E are the Arrhenius pre-exponential factor, temperature exponent and activation energy respectively. R is the gas constant and T is the temperature. θ_j is a property of the particle which is raised to some power p_j and J is the number of soot particle properties used in the rate expression. C_i is the concentration of gas-phase species i, ν_i is the forward stoichiometric coefficient of species i and I is the number of reactant species for the reaction. SWEEP has pre-programmed variables for θ^p, these are:

1. Volume (cm^3).
2. Mass (g).
3. Inverse mass to the power one half. ($g^{-0.5}$).
4. Collision diameter (cm).
5. Collision diameter squared (cm^2).
6. Inverse collision diameter (cm^{-1}).
7. Inverse collision diameter squared (cm^{-2}).
8. Surface area (cm^2).
9. Active surface area (cm^2).
10. Collision diameter squared times inverse mass to the power one half ($cm^2 g^{-0.5}$).

SWEEP is optimised to use these properties, reactions defined in the mechanism which do not use these properties will take substantially longer to solve. This is not recommended.

5.3 Condensation reactions

Condensation reactions in SWEEP are modelled as a free-molecular collision between a single gas-phase species and a soot particle. The rate (per particle) takes the form:

$$R_{cond} = \eta 2.2 C \sqrt{\frac{\pi k_B T}{2m}} \left(d^2 + 2 d d_p + d_p^2\right) \tag{7}$$

Where C, m and d are the gas-phase concentration (mol/cm^3), mass (g) and collision diameter (cm) of the condensing species respectively. d_p is the collision diameter of the particle. η is the efficiency of collision.

5.4 Coagulation

Coagulation is modelled in SWEEP using the Smoluchowski coagulation equation. This takes the form:

$$\frac{d}{dt} n(x) = \sum_{y=1}^{x-1} \beta(x-y, y) n(x-y) n(y) - \sum_{y=1}^{\infty} \beta(x, y) n(x) n(y) \tag{8}$$

$n(x)$ is the number density of particles of type x and $\beta(x, y)$ is the coagulation kernel of particles of types x and y. A discussion of the coagulation kernel in SWEEP is given by Patterson *et al.* (2006b). The forms of the free-molecular and slip-flow kernels have already been presented in the discussion of inception reactions.

5.5 Ensemble Rates

The total rates for the entire soot ensemble can be found by summing the per particle rates for each particle. For the stochastic simulation this will give you the rate per sample volume, that is the volume which would contain the number of particles in the ensemble. To get the rates per unit volume these rates must be divided by $V_{smp} = \frac{N}{M_0}$ where V_{smp} is the sample volume, N is the number of particles in the ensemble and M_0 is the particle number density.

6 Particle Models

SWEEP has two particle models programmed; the spherical particle model and the surface-volume model. The spherical particle model assumptions are:

1. Particles are perfect spheres and can be described by one size coordinate: volume.
2. Coagulation by total coalescence and does not conserve surface area.

The surface-volume model assumptions are:

1. Particles form aggregate structures and are described by two size coordinates: volume and surface area.
2. Coagulation by point contact and therefore conserves surface area.
3. All surface processes act to increase the sphericity of a particle. That is, particles become rounder due to surface growth and removal.

7 Settings Input File

The settings input file defines the settings of the solver at run time which are not set elsewhere. It must have the filename **sweep.inp** and be a plain text file.

Settings are set in the input file by writing a keyword (listed below) followed by the values. Only one settings may be written per line.

7.1 CHEMFILE keyword

Use this keyword to set the file name of the chemistry profile input.

```
CHEMFILE filename
```

filename – File name of chemistry profile.

7.2 MECHFILE keyword

Use this keyword to set the file name of the mechanism file.

```
MECHFILE filename
```

filename – File name of mechanism input.

7.3 RUNS keyword

Use this keyword to define how many runs to perform in order to get an averaged result.

```
RUNS num
```

num – Number of runs to perform. Positive integer.

7.4 STARTT keyword

Sets the start time of the simulation. This value must lie in the range given in the chemistry profile input.

```
STARTT time
```

time – Start time of the simulation. Real.

7.5 TIME keyword

Sets a stop time point for the simulation and gives the number of output steps to produce up to that time. At each time point the particle size list is output from SWEEP. Up to ten time points may be specified in the input file (one per line) and they must be given in chronological order.

```
TIME tim steps
```

tim – Stop time. Real.

steps – Number of output points up to the given time. Positive integer.

7.6 PCOUNT keyword

Sets the maximum stochastic particle count.

```
PCOUNT num
```

num – Maximum particle count. Positive integer, power of 2.

7.7 MAXM0 keyword

Used to set the internal scaling of SWEEP. For best results it should be set to the maximum expected number density of the simulation. If it is set too low then SWEEP has to remove particles from the ensemble, which leads to potential errors in volume fraction and can increase computation time. If it is set too high (more than one third above the maximum expected value) then the particle doubling algorithm will not work and there will probably be insufficient particles to get a statistically meaningful solution. In order to estimate this parameter it is suggested to select a value larger than expected (10^12 might be a good initial estimate) and run the simulation, noting the largest actual M_0 value observed. Then edit the input file with a value slightly higher (about 5%) than the true value.

```
MAXM0 num
```

num – The maximum expected number density. Real.

7.8 ACTSURFMODEL keyword

Used to tell SWEEP how the active surface model should be used. The options are: "profile" for the active surface fraction defined in the chemistry profile; "const" for a constant active surface and "abf" to use the ABF correlation for active surface fraction.

```
ACTSURFMODEL model value
```

model – Name of the model to use. One of "profile", "const" or "abf".

value – If the "const" active surface model is used then this value is used to supply the constant fraction. If it is omitted then the fraction is assumed to be one.

7.9 OUTPUTFILE keyword

Provides the name of the output files for the simulation.

```
OUTPUTFILE filename
```

filename – Name of the output files.

7.10 PMODEL keyword

Selects the particle model used for the simulation. If this keyword is omitted from the file then SWEEP assumes the spherical particle model.

```
PMODEL model
```

model – Name of the particle model to use. Accepted values are 'spherical" for the spherical particle model and 'surfvol" for the surface volume model.

7.11 LOWSTAT keyword

Sets the smallest diameter of particle that is summed for ensemble statistics.

`LOWSTAT value`

value – Smallest diameter particle in cm to be summed for ensemble statistics.

7.12 HIGHSTAT keyword

Sets the largest diameter of particle that is summed for ensemble statistics.

`HIGHSTAT value`

value – Largest diameter particle in cm to be summed for ensemble statistics.

8 Mechanism Input File (XML)

The particle mechanism for SWEEP is provided as an XML file. If you are unfamiliar with XML then it is advised you read up about it before reading this section. There are plenty of internet references and books on the subject, for example: http://www.w3schools.com/xml/. Following is a description of all the valid XML elements that can be used in a mechanism file and their contexts. An example mechanism file is provided in the appendix.

8.1 mechanism element

The root element in a mechanism file is the **mechanism** element and it has the following form:

```
<mechanism name=""></mechanism>
```

name – The name of the mechanism. This attribute is optional.

All other elements are written inside the **mechanism** element.

8.2 model element

The **model** element tells SWEEP to use a particular model when solving. An example of a model is how active surface is treated. The **model** element has the following form:

```
<model id="" type=""/>
```

id – Unique identifier for the model to activate.

type – Additional information about the model if more than one option is available.

8.2.1 Accepted model elements

Use the HACA model for calculation of radical site fraction (Frenklach and Wang, 1994).

```
<model id="haca"/>
```

Use the ABF correlation for active site fraction (alpha).

```
<model id="actsurf" type="abf"/>
```

8.3 particle element

The **particle** element defines a symbol used to represent a particle type in the XML file. To define more than one particle type include this element multiple times. The **particle** element has the following form:

```
<particle id=""/>
```

id – Symbol used to define the particle type.

8.4 component element

The **component** element takes two forms in a mechanism file. If it stands alone from an **inception** or **reaction** element then it defines a component used to build particles. In this case it must contain a **density** and a **molwt** element also and takes the form:

```
<component id="">
    <density>dens_val</density>
    <molwt>molwt_val</molwt>
</component>
```

id – Symbol used to identify the particle component.

dens_val – The component density in mol/cm^3.

molwt_val – The component molecular weight in g/mol.

The second form of the **component** element is as part of an inception or reaction definition and is used to define the change in that component's value due to the process. In this case it takes the form:

```
<component id="" dx=""/>
```

id – Symbol of component to change for process.

dx – The integer amount by which to change the component.

8.5 density element

The **density** element is only used inside the component element and takes the form:

```
<density>dens_val</density>
```

dens_val – The component density in mol/cm^3.

8.6 molwt element

The **molwt** element is only used inside the component element and takes the form:

```
<molwt>molwt_val</molwt>
```

molwt_val – The component molecular weight in g/mol.

8.7 track element

The **track** element comes in two flavours. First it is used to define a tracking variable which is summed per particle. Tracking variables are different from components because they are not used to calculate particle properties. A tracking variable is a real quantity and is defined by:

```
<track id=""/>
```

id – Symbol used to represent the tracking variable.

When the **track** element is used inside an **inception** or **reaction** element then it defines the change in that variable due to the process, and takes the form:

```
<track id="" dx=""/>
```

id – Symbol representing the tracking variable to change.

dx – The real amount by which to change the variable.

8.8 inception element

The **inception** element contains all the information required to define a particle inception process. Typically an **inception** element will take the form:

```
<inception name="">
    <formula>icn_form</formula>
    <reactant id="" stoich="" m="" d=""/>
    <reactant id="" stoich="" m="" d=""/>
    <product id="" stoich=""/>
    <particleout id=""/>
    <component id="" dx=""/>
    <track id="" dx=""/>
</inception>
```

name – Name of the inception reaction.

icn_form – Formula of the inception. This is optional and provided as a reference tool for the user.

An **inception** element can contain a maximum of two reactants, and in any case the stoichiometry of the reactants must sum to two because they are modelled as two-body processes. An error is produced if the stoichiometry does not sum to two. Reactant masses and collision diameters must also be provided. See the **reactant** element description for more details.

An **inception** element may contain any number of **product** elements, including zero.

An inception element must contain one and only one **particleout** element which defines the type of particle produced by the process.

An **inception** element must contain at least one **component** element and any number of **track** elements. Any components or tracker variables not explicitly given will be assigned a value of zero when the program is run.

8.9 reaction element

The **reaction** element contains all the information required to define a particle process other than inception. Typically a **reaction** element will take the form:

```
<reaction type="" name="" defer="">
    <formula>rxn_form</formula>
    <reactant id="" stoich="" m="" d=""/>
    <product id="" stoich=""/>
    <particlein id=""/>
    <particleout id=""/>
    <A>arrn_A</A>
    <n>arrn_n</n>
    <E>arrn_E</E>
    <particleterm id="" power=""/>
    <component id="" dx=""/>
    <track id="" dx=""/>
</reaction>
```

type – The type of single particle process. Accepted types are ”surface” and ”condensation”. These are explained later on.

name – Name of the reaction.

defer – true/false flag telling SWEEP whether this process should be simulated using LPDA (”true”) or DSA (”false”).

rxn_form – Formula of the reaction. This is optional and provided as a reference tool for the user.

8.9.1 surface type reactions

A **reaction** element of type "surface" can contain any number **reactant** elements and the reactant masses and collision diameters are not required. See the **reactant** element description for more details.

A **reaction** element of type "surface" may contain any number of **product** elements.

A **reaction** element of type "surface" must contain one and only one **particlein** element, which defines the type of particle to which the process occurs.

A **reaction** element of type "surface" may contain a maximum of one **particleout** element, which defines the type of particle output from the process. If no **particleout** element is given then the output particle is the same type as the input particle. If the output particle is of a different type from the input particle then the process may not be deferred, and the defer attribute is ignored.

A **reaction** element of type "surface" may contain elements which define the Arrhenius rate coefficients; **A, n, E**. If these elements are omitted then the default values are used. See the definitions of these elements for more details.

A **reaction** element of type "surface" must contain a **particleterm** element which defines how the rate expression depends on the particle properties. See the definition of the **particleterm** element for more details.

8.9.2 condensation type reactions

A **reaction** element of type "condensation" must contain only one **reactant** element and the reactant masses and collision diameters must be given. See the **reactant** element description for more details.

A **reaction** element of type "condensation" may contain any number of **product** elements.

A **reaction** element of type "condensation" must contain one and only one **particlein** element, which defines the type of particle to which the process occurs.

A **reaction** element of type "condensation" may contain a maximum of one **particleout** element, which defines the type of particle output from the process. If no **particleout** element is given then the output particle is the same type as the input particle. If the output particle is of a different type from the input particle then the process may not be deferred, and the defer attribute is ignored.

A **reaction** element of type "condensation" may contain an **A** element to define a rate constant which is interpreted as a reaction efficiency. No other Arrhenius rate elements are required.

A **reaction** element of type "condensation" may not contain a **particleterm** element.

A **reaction** element of type "condensation" may contain any number of **component** and **track** elements. Any components or tracker variables not explicitly given will be assigned a value of zero when the program is run.

8.10 formula element

The **formula** element is used inside an **inception** or **reaction** element and is provided only as a reference for the user. It is not used by SWEEP. It takes the form:

```
<formula>rxn_form</formula>
```

rxn_form – The reaction formula string.

8.11 reactant element

The **reactant** element is used inside the **inception** and **reaction** elements to define a reactant chemical species in the gas-phase. It takes the form:

```
<reactant id="" stoich="" m="" d=""/>
```

id – The symbol of the reactant species. This must match the symbols provided to SWEEP via the input file.

stoich – The integer stoichiometric coefficient of the reactant species. There may be some restrictions on this value depending on the context. See definitions of the **inception** and **reaction** elements for more details.

m – The mass of the reactant species in g. This is required for inceptions and condensation type reactions, otherwise it can be omitted.

d – The collision diameter of the reactant species in cm. This is required for inceptions and condensation type reactions, otherwise it can be omitted.

8.12 product element

The **product** element is used inside the **inception** and **reaction** elements to define a product chemical species to the gas-phase. It takes the form:

```
<product id="" stoich=""/>
```

id – The symbol of the product species. This must match the symbols provided to SWEEP via the input file.

stoich – The integer stoichiometric coefficient of the product species.

8.13 particlein element

The **particlein** element is used inside the **inception** and **reaction** elements to define the reactant particle type. It takes the form:

```
<particlein id=""/>
```

id – The symbol representing the reactant particle type.

8.14 particleout element

The **particleout** element is used inside the **inception** and **reaction** elements to define the product particle type. It takes the form:

```
<particleout id=""/>
```

id – The symbol representing the product particle type.

8.15 A, n and E elements

The **A**, **n** and **E** elements are used inside the **inception** and **reaction** elements to define the Arrhenius rate parameters for expressions of the form $AT^n \exp(-E/RT)$. They take the form:

```
<A>arrn_A</A>
```

```
<n>arrn_n</n>
```

```
<E units="">arrn_E</E>
```

arrn_A – Arrhenius pre-exponential coefficient.

arrn_n – Arrhenius temperature exponent.

arrn_E – Arrhenius activation energy.

units – Units of the activation energy. Permitted values are "si" for energies in Joules and "cal" for energies in kilo-calories.

8.16 particleterm element

The **particleterm** element is used inside the **inception** and **reaction** elements to define how the rate expression depends on the particle properties. It takes the form:

```
<particleterm id="" power=""/>
```

id – ID of the particle property used to calculate the rate expression. Permitted values are "d" = collision diameter (cm), "s" = surface area (cm^2), "as" = active surface area (cm^2) and "v" = volume (cm^3).

power – The exponent to which to raise the particle property. Only certain combinations of properties and powers are included in SWEEP and if a property and power combination is chosen which is not pre-programmed then this will have a significant effect of computation time.

8.17 coagulation element

The **coagulation** element is used to define the resultant particle type of a coagulation if more than one particle type has been defined. It will take the form:

```
<coagulation name="">
    <formula>coag_form</formula>
    <particlein id=""/>
    <particlein id=""/>
    <particleout id=""/>
</coagulation>
```

name – Reference name of the coagulation.

coag_form – The formula of the coagulation. This is provided for user reference only and is not used by SWEEP.

The **coagulation** element must contain two **particlein** elements to define the types of particles which are coagulating.

The **coagulation** element must contain only one **particleout** element to define the resultant particle type of the coagulation.

If a possible coagulation exists for which no **coagulation** element has been defined then the resultant particle type is the first type defined in the mechanism, **not** that defined in the coagulation!

9 Chemistry Profile File

The chemistry profile file is a text file with tab- or comma-separated data. The first line of the file should contain two values. The first is the number of number of points in the chemistry profile and the second is the number of columns in the file *excluding* the time column. The second line in the file contains the header row for the subsequent profile. There are four reserved column headings, three of which must be included in the file. These are:

1. *Time* – Denotes the column of time coordinates.
2. *T* – Denotes the column of gas-phase temperature.
3. *P* – Denotes the column of gas-phase pressure.
4. *Alpha* – Denotes the column of soot active site fraction. This column is optional.

The other columns should be headed by the names of the pertinent gas-phase species. These names should match those used in the mechanism XML file, otherwise the program fail.

The units of the chemistry file are $seconds$ for time, $Kelvin$ for temperature, bar for pressure and mol/cm^3 for species concentrations. Alpha is a dimensionless quantity with a value between zero and one.

10 Output Files

Data for each run is stored in a numbered binary file with the name supplied in the settings file. These file are given the *.dat* extension. Once all runs have been completed these binary files are processed and the averaged data is output to CSV file along with the 99% confidence intervals. The output file contains statistics and information about the particle population along with the process rates at each output point. The statistics output, in order, are:

1. Stochastic particle count.
2. Number density (cm^{-3}).
3. First size moment ($\mathrm{atoms}/\mathrm{cm}^{-3}$).
4. Particle volume fraction.
5. Particle mass concentration ($\mathrm{g}/\mathrm{cm}^{3}$).
6. Total particle surface area ($\mathrm{cm}^{2}/\mathrm{cm}^{3}$).
7. Average first size moment per particle.
8. Average particle volume (cm^{3}).
9. Average particle mass (g).
10. Average particle surface area (cm^{2}).
11. Active surface fraction.
12. Average particle diameter (nm).
13. Sample volume (cm^{3}).
14. Ensemble scaling factor.
15. Second size moment ($\mathrm{atoms}^{2}/\mathrm{cm}^{-3}$).
16. Third size moment ($\mathrm{atoms}^{3}/\mathrm{cm}^{-3}$).
17. Fourth size moment ($\mathrm{atoms}^{4}/\mathrm{cm}^{-3}$).
18. Fifth size moment ($\mathrm{atoms}^{5}/\mathrm{cm}^{-3}$).
19. Sixth size moment ($\mathrm{atoms}^{6}/\mathrm{cm}^{-3}$).

Additionally, the particle size list (PSL) is output at the end of each time interval as defined in the settings file. These are also written to binary file for each run and then post-processed on completion of the program to provide a single CSV file with a list of all particles for all runs. The PSL contains the following information for each particle:

1. Relative weighting.
2. Volume (cm^3).
3. Surface area (cm^2).
4. Diameter (nm).
5. Particle type number.
6. Number of each particle component.
7. Values for all tracking variables.

11 Example Settings File

```
CHEMFILE jw1.69.dat
MECHFILE abfmech.xml
RUNS 10
STARTT 0.0D0
TIME 4.0D-4 400
TIME 1.4D-3 200
PCOUNT 2048
MAXM0 6.0e8
ACTSURFMODEL profile
OUTPUTFILE test
```

12 Example Chemistry Profile

This is an example chemistry profile with the gas-phase species concentrations omitted for clarity.

```
3,11
Time,T,P,Alpha,C2H2,H2,H,O2,OH,H2O,CO,A4
0.00,500,6,0.8,...
0.05,500,6,0.8,...
0.10,500,6,0.8,...
```

13 Example Mechanism File

```
<?xml version="1.0" encoding="ISO-8859-1"?>
```

```
<mechanism name="abf soot">

    <!-- Choose which models to use -->

    <model id="haca"/>
    <model id="actsurf" type="abf"/>

    <!-- Define particle types and components. -->

    <particle id="[X]"/>

    <component id="C">
        <density>1.8</density>
        <molwt>12.011</molwt>
    </component>

    <!-- Tracker variables -->

    <track id="H"/>

    <!-- Inceptions -->

    <inception name="Pyrene inception">
        <formula>A4 + A4 to [X32]</formula>
        <reactant id="A4" stoich="2" m="3.19e-22" d="0.79e-7"/>
        <particleout id="[X]"/>
        <component id="C" dx="32"/>
        <track id="H" dx="20"/>
    </inception>

    <!-- Surface reactions (single particle events) -->

    <reaction type="surface" name="OH oxidation" defer="true">
        <formula>[X] + OH to [X-1] + CO + H</formula>
        <reactant id="OH" stoich="1"/>
        <product id="CO" stoich="1"/>
        <product id="H" stoich="1"/>
        <A>2.1693357e26</A>
        <n>0.5</n>
        <E>0.0</E>
        <particleterm id="d" power="2.0"/>
        <component id="C" dx="-1"/>
    </reaction>

    <reaction type="surface" name="C2H2 addition" defer="true">
```

```
        <formula>[X] + C2H2 to [X+2] + H</formula>
        <reactant id="C2H2" stoich="1"/>
        <product id="H" stoich="1"/>
        <particlein id="[X]"/>
        <particleout id="[X]"/>
        <A>8.0E7</A>
        <n>1.56</n>
        <E units="cal">3.8E0</E>
        <particleterm id="as" power="1.0"/>
        <component id="C" dx="2"/>
        <track id="H" dx="+1"/>
    </reaction>

    <reaction type="surface" name="O2 oxidation" defer="true">
        <formula>[X] + O2 to [X-2] + 2CO</formula>
        <reactant id="O2" stoich="1"/>
        <product id="CO" stoich="2"/>
        <A>2.2e12</A>
        <n>0.0</n>
        <E units="cal">7.5</E>
        <particleterm id="as" power="1.0"/>
        <component id="C" dx="-2"/>
    </reaction>

    <!-- Condensation reactions (single particle events) -->

    <reaction type="condensation" name="A4 condensation" defer="true">
        <formula>[X] + A4 to [X+16]</formula>
        <reactant id="A4" stoich="1" m="3.19e-22" d="0.79e-7"/>
        <component id="C" dx="+16"/>
        <track id="H" dx="+10"/>
    </reaction>

    <!-- Coagulation rules. -->

    <coagulation name="default">
        <formula>[X] + [X] to [X]</formula>
        <particlein id="[X]"/>
        <particlein id="[X]"/>
        <particleout id="[X]"/>
    </coagulation>

</mechanism>
```

References

Frenklach, M. & Wang, H. (1994). *Detailed Mechanism and Modeling of Soot Particle Formation*, pp. 162–192). Springer Verlag, New York.

Patterson, R., Singh, J., Balthasar, M., Kraft, M. & Norris, J. (2006a). The Linear Process Deferment Algorithm: A new technique for solving population balance equations. *SIAM Journal on Scientific Computing* (28), 303–320.

Patterson, R., Singh, J., Balthasar, M., Kraft, M. & Wagner, W. (2006b). Extending stochastic soot simulation to higher pressures. *Combust. Flame* (145), 638–642.